UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/699,486	10/31/2003	Eric Anderson	200207252-1	3149

22879   7590   10/30/2007
HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER
RADTKE, MARK A

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
10/30/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 August 2007*.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,2 and 4-24* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2 and 4-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

DETAILED ACTION

Remarks

1.   In response to communications filed on 14 August 2007, claim(s) 1, 2, 6, 9, 12-14

and 19-24 is/are amended per Applicant's request. Therefore, claims 1-2 and 4-24 are

presently pending in the application, of which, claim(s) 1, 12 and 19-24 is/are presented

in independent form.

2.   In light of Applicant's amendments, the formatting problems related to 37 CFR

1.121(c)(2) have been corrected. The newly amended claims conform to standard

practice since the bracketed portions have been deleted. Applicant's amendments have

necessitated new grounds of rejection.

Claim Rejections - 35 USC § 103

3.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set
> forth in section 102 of this title, if the differences between the subject matter sought to be patented and
> the prior art are such that the subject matter as a whole would have been obvious at the time the
> invention was made to a person having ordinary skill in the art to which said subject matter pertains.
> Patentability shall not be negatived by the manner in which the invention was made.

4.   Claims 1-2, 4-15, and 17-24 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Verma et al. (U.S. Pat. No. 6,856,993), in view of Berliner ("CVS II:

Parallelizing Software Development" by B. Berliner, Proceedings of the USENIX Winter

1990 Technical Conference, available online at

http://citeseer.ist.psu.edu/berliner90cvs.html) and further in view of Deshayes (U.S. Pat.

No. 6,047,294).

As to claim 1, Verma et al. teaches a method of creating a filesystem with

transaction based functionality (see Abstract), comprising:

receiving an indicator to initiate a transaction for files stored in one or more

portions of the filesystem (see column 10, lines 8-10, "mark the thread/process as

transacted" and column 10, lines 20-24, "copyFile");

processing the text-based commands written to the control text file (see column

2, lines 57-59 and column 3, lines 3-6); and

operating on one or more portions of the pseudo-filesystem within a transaction

according to the text-based commands (see column 3, lines 3-6).

Verma et al. does not explicitly teach creating a control text file that provides a

textual filesystem interface and receives text-based commands to operate on the

pseudo-filesystem.

However, Berliner teaches creating a control text file that provides a textual

filesystem interface and receives text-based commands to operate on the pseudo-

filesystem (See section 2.2, "Tracking Third-Party Source Distributions", "**checkin**

program". Berliner anticipates the use of scripts, which are equivalent to "a control text

file". Furthermore, the use of scripts to automate certain tasks is extremely well-known

in the art of Unix systems programming. See, for example, "Running Arbitrary Scripts

Under CVS" by J. Vesperman. Furthermore, to an application, the Unix command line ("stdin") is indistinguishable from a text file).

Therefore, it would have been obvious to one having ordinary skill in the relevant art at the time the invention was made to have modified Verma et al. by the teaching of Berliner because "other operating systems and/or file systems may implement and benefit from the present invention" (see Verma et al., column 6, lines 17-19).

Verma et al., as modified, still does not teach duplicating the filesystem within a pseudo-filesystem.

However, Deshayes teaches duplicating the filesystem within a pseudo-filesystem (see Abstract, "a virtual disk partition, may be backed up at a physical level from a primary storage device").

Therefore, it would have been obvious to one of ordinary skill in the relevant art at the time the invention was made to have further modified Verma et al., as modified, by the teaching of Deshayes because all the claimed elements were known in the prior art and one skilled in the art could have combined by known methods with no change in their respective functions, and the combination would have yielded predictable results to one of ordinary skill in the art at the time the invention was made. Furthermore, it would provide the benefit of solving the problem of having "to change the operating system, or the application programs, every time a change is made to physical storage" (see Deshayes, column 1, lines 30-32).

As to claim 2, <u>Verma et al.</u>, as modified, teaches wherein the duplicating is performed lazily (see column 2, lines 59-65 and column 23, "Deferred Redo Alternative") in order to reduce processing impact on the filesystem (This portion of the claim is considered "intended use" and will not be given patentable weight. The effect of "reducing processing impact" is merely a benefit of using the invention and does not functionally relate to the claimed invention).

As to claim 4, <u>Verma et al.</u>, as modified, teaches further comprising:

completing the transaction upon receipt of a text-based command associated with terminating the transaction (see column 8, lines 26-28).

As to claim 5, <u>Verma et al.</u>, as modified, teaches wherein the text-based commands include functional equivalent commands associated with terminating the transaction (see column 7, lines 23-26, "aborted") and selected from a set of commands for performing one of the following functions: delete directory (see column 17, lines 3-7), delete filesystem (see column 17, lines 3-7, "recursive delete"), and abort (see column 7, lines 23-26).

As to claim 6, <u>Verma et al.</u>, as modified, teaches further comprising:

updating the filesystem with updates performed on the pseudo-filesystem when the transaction has completed (see column 8, lines 26-28).

As to claim 7, <u>Verma et al.</u>, as modified, teaches wherein the updates are performed upon receipt of an indication to commit the transaction (see column 8, lines 26-28).

As to claim 8, <u>Verma et al.</u>, as modified, teaches further comprising:

creating a status text file that provides text-based status results from operations performed on the pseudo-filesystem (see column 2, lines 57-59, "actual data write details of the transaction").

As to claim 9, <u>Verma et al.</u>, as modified, teaches wherein the indicator to initiate the transaction results from the creation of a directory within the pseudo-filesystem (see column 27, lines 64-67).

As to claim 10, <u>Verma et al.</u>, as modified, teaches wherein the transaction ensures atomic updates to the filesystem in accordance with modifications made to the pseudo-filesystem and related files during the transaction (see column 6, lines 24-26).

As to claims 11 and 18, <u>Verma et al.</u>, as modified, teaches wherein a user assists in reconciliation of conflicts between updates in the pseudo-filesystems (See column 29, lines 37-45. Depending on when the non-transacted user releases the resource, a file handle in conflict will not be deleted, thus resolving a resource conflict).

As to claim 12, <u>Verma et al.</u> teaches a method of interfacing with a filesystem (see Abstract) comprising:

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 13, <u>Verma et al.</u>, as modified, teaches creating an entire copy of the filesystem (See Examiner's comments regarding claim 1. See <u>Deshayes</u>, Abstract, "a virtual disk partition, may be backed up at a physical level from a primary storage device");

mounting the entire copy of the filesystem under the pseudo-filesystem (see <u>Deshayes</u>, column 12, lines 54-58, "mounting or importing virtual volumes").

As to claim 14, <u>Verma et al.</u>, as modified, teaches creating a textual interface;

receiving the text-based command from a user into the textual interface (see column 10, lines 23-24, "command line batch scripts").

As to claim 15, <u>Verma et al.</u>, as modified, teaches wherein receiving a text-based command includes functional equivalent commands selected from a set including: change root directory (The "mount" command is all well-known command in NTFS. Mount points can be partitions or folders within an existing partition. See http://support.microsoft.com/?kbid=205524), select concurrency control type (See column 6, lines 56-59. Any kind of concurrency control system can be used via

interfaces), select isolation level (See column 6, lines 48-51. Processes, file handles or files must be selected before they are treated as transactional operations. Disabling or enabling transactions is a selection of isolation level.), commit transaction (see column 8, lines 26-28), and abort transaction (see column 7, lines 23-26).

As to claim 17, Verma et al., as modified, teaches wherein determining the one or more data dependencies includes using lock-based concurrency control (LBCC) to control pending read and write operations to the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively (see column 11, line 49 – column 12, line 18).

As to claim 19, Verma et al. teaches a computer program product for creating a filesystem with transaction based functionality, tangibly stored on a computer-readable medium, comprising instructions operable to cause a programmable processor (see Abstract) to:

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 20, Verma et al. teaches a computer program product for interfacing with a filesystem, tangibly stored on a computer-readable medium, comprising instructions operable to cause a programmable processor (see Abstract) to:

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 21, Verma et al. teaches an apparatus that creates a filesystem with transaction based functionality (see Abstract) comprising:

a processor (see figure 1, element 21);

a memory (see figure 1, element 25) having instructions capable of being executed on the processor...

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 22, Verma et al. teaches an apparatus that interfaces with a filesystem (see Abstract), comprising:

a processor (see figure 1, element 21);

a memory (see figure 1, element 25) having instructions capable of being executed on the processor...

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 23, Verma et al. teaches an apparatus for creating a filesystem with transaction based functionality (see Abstract), comprising:

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

As to claim 24, <u>Verma et al.</u> teaches an apparatus for interfacing with a filesystem (see Abstract), comprising:

For the remaining steps of this claim applicant(s) is/are directed to the remarks and discussions made in claim 1 above.

5.   Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over <u>Verma et al.</u>, as modified, as applied to claim 12 above, and further in view of <u>Kung et al.</u> ("On optimistic methods for concurrency control", ACM Transactions on Database Systems (TODS), vol. 6, issue 2, pages 213-226. Published June 1981).

As to claim 16, <u>Verma et al.</u>, as modified, still does not teach wherein determining the one or more data dependencies includes using optimistic concurrency control (OCC) to control pending read and write operations to the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively.

<u>Kung et al.</u> teaches wherein determining the one or more data dependencies includes using optimistic concurrency control (OCC) to control pending read and write operations to the pseudo-filesystem, the filesystem and one or more corresponding files associated with the pseudo-filesystem and filesystem respectively (see Abstract).

Therefore, it would have been obvious to one of ordinary skill in the relevant art at the time the invention was made to have modified Verma et al., as modified, by the teaching of Kung et al. for the benefit of providing an external transaction service (See Verma et al., column 6, lines 59-64, where one type of transaction service, MS-DTC, is suggested. Furthermore, Examiner notes that there are 171 citations listed on the ACM Portal, indicating that the method is well-known in the art).

Response to Arguments

6.   Applicant's arguments filed on 14 August 2007 with respect to the rejected claims in view of the cited references have been fully considered but are not deemed persuasive.

In response to Applicant's arguments that the combination of references does not "teach or even suggest two filesystems", the arguments have been fully considered but are not deemed persuasive. Applicant notes that the use of "filesystem" in the instant specification is "consistent with the plain meaning of the term", however the point is moot. Applicant uses the new, non-standard term "pseudo-filesystem" throughout the specification and claim. The term is undefined both in the specification and in the claims. Therefore, it is appropriate to apply the broadest reasonable interpretation. One such interpretation is anything that acts like or in place of a filesystem. The Microsoft Computer Dictionary, Fifth Edition defines "file system" as "the overall structure in which

files are named, stored, and organized. *A file system consists of files, directories, or folders, and the information needed to locate and access these items.* The term can also refer to the portion of an operating system that translates requests for file operations from an application program into low-level, sector-oriented tasks that can be understood by the drivers controlling the disk drives" (emphasis added). CVS is an application for managing files. Files can be checked in or checked out. These operations are equivalent to basic file system operations of saving, modifying and opening files. CVS supports directory structures. So, CVS acts like a filesystem even though is not a file system *per se*. Thus, it is a "pseudo-filesystem".

Additional References

7.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

The following patents are cited to further show the state of art with respect to file systems in general:

Doc. No.	Assigned to
US 6470345 B1	Doutre; Edward et al.
US 5001628 A	Johnson; Donavon W. et al.
US 5870757 A	Fuller; Billy J.
US 6985914 B2	Venkatesh; Dinesh et al.
US 5991753 A	Wilde; Michael J.
US 6606685 B2	Huxoll; Vernon F.
US 7076685 B2	Pillai; Ananthan K. et al.

"Virtual Swap Space in SunOS" by H. Chartock and P. Snyder.

"JFFS: The Journalling Flash File System" by D. Woodhouse.

"The LFS Storage Manager" by M. Rosenblum and J.K. Ousterhout.

"Beating the I/O Bottleneck: A Case for Log-Structured File Systems" by J.

Ousterhout and F. Douglis.

"Coda: A Highly Available File System for a Distributed Workstation

Environment" by M. Satyanarayanan et al.

"Vnodes: An Architecture for Multiple File System Types in Sun UNIX" by S.R.

Kleiman.

"Ivy: A Read/Write Peer-to-Peer File System" by A. Muthitacharoen et al.

Conclusion

8.   Applicant's amendment necessitated the new ground(s) of rejection presented in

this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a).

Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

9.   Any inquiry concerning this communication or earlier communications should be directed to the examiner, Mark A. Radtke. The examiner's telephone number is (571) 272-7163, and the examiner can normally be reached between 9 AM and 5 PM, Monday through Friday.

If attempts to contact the examiner are unsuccessful, the examiner's supervisor, Jeffrey Gaffin, can be reached at (571) 272-4146.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to Customer Service at (800) 786-9199.

maxr
***

25 October 2007

Nerveen Abel-Jalil
Primary Examiner